Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1-80. (canceled)

81. (new) A method comprising:

in a platform with a processor and a memory, configuring the processor to run in an isolated execution mode within a ring 0 operating mode, wherein the processor also supports one or more higher ring operating modes, as well as a non-isolated execution mode within at least the ring 0 operating mode;

configuring the platform to establish an isolated memory area in the memory and a non-isolated memory area in the memory, wherein the platform does not allow access to the isolated memory area if the processor is not in the isolated execution mode;

executing a processor executive on the processor, with the processor running in the isolated execution mode;

loading an operating system (OS) executive into the isolated memory area, the OS executive to manage at least a subset of an OS to run on the platform;

verifying the OS executive, using the processor executive; and

after verifying the OS executive, launching the OS executive, the launching of the OS executive performed by the processor executive.

82. (new) The method of claim 81, wherein the operation of verifying the OS executive comprises:

verifying the OS executive during a process of booting the platform.

83. (new) The method of claim 82, further comprising:

logging a processor executive identifier during the process of booting the platform; and

logging an OS executive identifier during the process of booting the platform.

84. (new) The method of claim 81, further comprising:

loading the processor executive into the isolated memory area; and

verifying the processor executive, based at least in part on a processor executive manifest.

85. (new) The method of claim 81, wherein the operation of launching the OS executive comprises:

launching the OS executive to run in the isolated execution mode.

86. (new) The method of claim 81, further comprising:

switching from the isolated execution mode to the non-isolated execution mode;

loading an OS kernel into non-isolated memory; and

executing the OS kernel in the non-isolated mode of the processor.

87. (new) The method of claim 81, wherein:

the platform comprises a platform key (PK); and

verification of the OS executive is based at least in part on the PK.

88. (new) The method of claim 87, wherein the PK comprises a symmetric encryption/decryption key that is substantially uniquely assigned to the platform.

89. (new) The method of claim 87, further comprising:

generating a processor executive key (PEK), based at least in part on a processor executive identifier and the PK.

90. (new) The method of claim 89, further comprising:

generating a binding key (BK), based at least in part on the PEK.

91. (new) The method of claim 90, further comprising:

generating an OS executive key (OSEK), based at least in part on an OS executive identifier and the BK.

92. (new) The method of claim 81, wherein the OS executive manages at least the subset of the OS by performing operations comprising:

loading a module into the isolated memory area;

managing paging in the isolated memory area; and

interfacing with an OS kernel.

93. (new) The method of claim 81, wherein the OS executive performs operations comprising:

loading a module into the isolated memory area, the module selected from a group consisting of an application module, an applet module, and a support module.

94. (new) The method of claim 93, wherein the OS executive performs further operations comprising:

generating an applet key associated with the applet module.

95. (new) The method of claim 94, wherein the OS executive generates the applet key based at least in part on an OS executive key and an applet identifier identifying the applet module.

96. (new) The method of claim 81, further comprising:

executing an isolated create instruction during a process of booting the platform, wherein execution of the isolated create instruction launches an atomic sequence of operations, the atomic sequence being non-interruptible, the atomic sequence of operations comprising:

reading a thread count register in a chipset to determine if the processor is the first processor in the isolated execution mode;

configuring the processor in the isolated execution mode;

loading a processor executive handler into the isolated memory area.

verifying the loaded processor executive handler; and

transferring control to the loaded processor executive handler.

97. (new) The method of claim 96, wherein the chipset includes at least one hub selected from a group consisting of a memory controller hub (MCH) and an input/output controller hub (ICH).

98. (new) An apparatus comprising:

a machine accessible medium; and

instructions encoded in the machine accessible medium, wherein the instructions, when executed in a platform featuring a processor and a memory, cause the platform to perform operations comprising:

configuring the processor to run in an isolated execution mode within a ring 0 operating mode, wherein the processor also supports one or more higher ring operating modes, as well as a non-isolated execution mode within at least the ring 0 operating mode;

establishing an isolated memory area in the memory and a non-isolated memory area in the memory, wherein the platform does not allow access to the isolated memory area if the processor is not in the isolated execution mode;

executing a processor executive on the processor, with the processor running in the isolated execution mode;

loading an operating system (OS) executive into the isolated memory area, the OS executive to manage at least a subset of an OS to run on the platform;

verifying the OS executive, using the processor executive; and

after verifying the OS executive, launching the OS executive, the launching of the OS executive performed by the processor executive.

99. (new) The apparatus of claim 98, wherein the operation of verifying the OS executive comprises:

verifying the OS executive during a process of booting the platform.

100. (new) The apparatus of claim 99, wherein the instructions cause the platform to perform further operations comprising:

logging a processor executive identifier during the process of booting the platform; and

logging an OS executive identifier during the process of booting the platform.

101. (new) The apparatus of claim 98, wherein the instructions cause the platform to perform further operations comprising:

loading the processor executive into the isolated memory area; and

verifying the processor executive, based at least in part on a processor executive manifest.

102. (new) The apparatus of claim 98, wherein the operation of launching the OS executive comprises:

launching the OS executive to run in the isolated execution mode.

103. (new) The apparatus of claim 98, wherein the instructions cause the platform to perform further operations comprising:

switching the processor from the isolated execution mode to the non-isolated execution mode;

loading an OS kernel into non-isolated memory; and

executing the OS kernel in the non-isolated mode of the processor.

104. (new) The apparatus of claim 98, wherein:

the platform comprises a platform key (PK); and

the platform verifies the OS executive, based at least in part on the PK.

105. (new) The apparatus of claim 104, wherein the instructions cause the platform to perform further operations comprising:

generating a processor executive key (PEK), based at least in part on a processor executive identifier and the PK.

106. (new) The apparatus of claim 105, wherein the instructions cause the platform to perform further operations comprising:

generating a binding key (BK), based at least in part on the PEK; and

generating an OS executive key (OSEK), based at least in part on an OS executive identifier and the BK.

107. (new) The apparatus of claim 98, wherein:

the instructions comprise the OS executive; and

the OS executive manages at least the subset of the OS by performing operations comprising:

loading a module into the isolated memory area;

managing paging in the isolated memory area; and

interfacing with an OS kernel.

108. (new) The apparatus of claim 98, wherein:

the instructions comprise the OS executive; and

the OS executive loads a module into the isolated memory area, the module selected from a group consisting of an application module, an applet module, and a support module.

109. (new) The apparatus of claim 108, wherein the OS executive generates an applet key associated with the applet module, the applet key based at least in part on an OS executive key and an applet identifier identifying the applet module.

110. (new) The apparatus of claim 98, wherein the instructions cause the platform to perform further operations comprising:

executing an isolated create instruction during a process of booting the platform, wherein execution of the isolated create instruction launches an atomic sequence of operations, the atomic sequence being non-interruptible, the atomic sequence of operations comprising:

reading a thread count register in a chipset to determine if the processor is the first processor in the isolated execution mode;

configuring the processor in the isolated execution mode;

loading a processor executive handler into the isolated memory area;

verifying the loaded processor executive handler; and

transferring control to the loaded processor executive handler.

111. (new) A system comprising:

a platform featuring memory and a processor, wherein the processor is capable of running in an isolated execution mode within a ring 0 operating mode, wherein the processor supports one or more higher ring operating modes, and wherein the processor supports a non-isolated execution mode within at least the ring 0 operating mode;

multiple machine accessible media in the platform, the multiple machine accessible media comprising at least non-volatile memory and storage within the processor; and

instructions encoded in at least one of the machine accessible media, wherein the instructions, when executed in the platform, cause the platform to perform operations comprising:

configuring the processor to run in the isolated execution mode;

establishing an isolated memory area in the memory and a non-isolated memory area in the memory, wherein the platform does not allow access to the isolated memory area if the processor is not in the isolated execution mode;

executing a processor executive on the processor, with the processor running in the isolated execution mode;

loading an operating system (OS) executive into the isolated memory area, the OS executive to manage at least a subset of an OS to run on the platform;

verifying the OS executive, using the processor executive; and

after verifying the OS executive, launching the OS executive, the launching of the OS executive performed by the processor executive.

112. (new) The system of claim 111, wherein the operation of verifying the OS executive comprises:

verifying the OS executive during a process of booting the platform.

113. (new) The system of claim 112, wherein the instructions cause the platform to perform further operations comprising:

logging a processor executive identifier during the process of booting the platform; and

logging an OS executive identifier during the process of booting the platform.

114. (new) The system of claim 111, wherein the instructions cause the platform to perform further operations comprising:

loading the processor executive into the isolated memory area; and

verifying the processor executive, based at least in part on a processor executive manifest.

115. (new) The system of claim 111, wherein the operation of launching the OS executive comprises:

launching the OS executive to run in the isolated execution mode.

116. (new) The system of claim 111, wherein the instructions cause the platform to perform further operations comprising:

switching the processor from the isolated execution mode to the non-isolated execution mode;

loading an OS kernel into non-isolated memory; and

executing the OS kernel in the non-isolated mode of the processor.

117. (new) The system of claim 111, wherein:

the system further comprises a platform key (PK); and

the platform verifies the OS executive, based at least in part on the PK.

118. (new) The system of claim 111, wherein he platform further comprises:

a chipset communicatively coupled to the processor;

an input/output controller hub in the chipset; and

a platform key (PK) stored in the input/output controller hub; and

wherein the platform verifies the OS executive, based at least in part on the PK.

119. (new) The system of claim 118, wherein the instructions cause the platform to perform further operations comprising:

generating a processor executive key (PEK), based at least in part on a processor executive identifier and the PK.

120. (new) The system of claim 119, wherein the instructions cause the platform to perform further operations comprising:

generating a binding key (BK), based at least in part on the PEK; and

generating an OS executive key (OSEK), based at least in part on an OS executive identifier and the BK.

121. (new) The system of claim 111, wherein:

the instructions comprise the OS executive; and

the OS executive manages at least the subset of the OS by performing operations comprising:

loading a module into the isolated memory area;

managing paging in the isolated memory area; and

interfacing with an OS kernel.

122. (new) The system of claim 111, wherein:

the instructions comprise the OS executive; and

the OS executive loads a module into the isolated memory area, the module selected from a group consisting of an application module, an applet module, and a support module.

123. (new) The system of claim 122, wherein the OS executive generates an applet key associated with the applet module, the applet key based at least in part on an OS executive key and an applet identifier identifying the applet module.

124. (new) The system of claim 111, wherein the instructions cause the platform to perform further operations comprising:

executing an isolated create instruction during a process of booting the platform, wherein execution of the isolated create instruction launches an atomic sequence of operations, the atomic sequence being non-interruptible, the atomic sequence of operations comprising:

reading a thread count register in a chipset to determine if the processor is the first processor in the isolated execution mode;

configuring the processor in the isolated execution mode;

loading a processor executive handler into the isolated memory area;

verifying the loaded processor executive handler; and

transferring control to the loaded processor executive handler.

Amendments to the Drawings:

The five attached replacement sheets include Figures 2, 3, 5, 7, and 10, revised as listed below:

- In Figure 2, in block 260, the label "Fused Key (FK)" has been changed to "Platform Key (PK)".
- In Figure 3, the label "FK 260" has been changed to "PK 260".
- In Figure 5, reference no. 544 has been added to the "Applet Key Combiner"
- In Figure 7, in block 720, the label "Fused Key" has been changed to "Platform Key".
- In Figure 10, in block 1030, the label "Fused Key" has been changed to "Platform Key".

All of these changes merely bring the drawings into correspondence with the Detailed Description. The drawing changes do not introduce any new matter. Approval and entry of these replacement drawings is respectfully requested.